Bachelor's Thesis

Bachelor's Degree in Industrial Technology Engineering

Control of a Two-Wheeled Line Tracking Robot Using a Complete Mechanical Model

Author: Judit Ruiz Cabo
Advisor: Arnau Dòria Cerezo
Announcement: April 2018

Escola Tècnica Superior
d'Enginyeria Industrial de Barcelona

Abstract

The aim of this project is to design the control of a two-wheeled line tracking robot used for a small-scale model of a platooning system.

Previous works have based the design in a mechanical model that only includes the kinematics of the robot. In this project, the model is completed by adding the dynamical behaviour and the controller is redesigned accordingly.

In order to do that, the obtained nonlinear model has been linearized and decoupled for a particular case. Two controllers have been designed using the root locus method for each of the subsystems, one of which needs a feed forward term in order to improve the stability range of the closed loop system. The decoupled system responds successfully for the desired range of velocities of the robot.

Finally, both the linear and the nonlinear models with the designed control algorithm have been simulated for different cases, using MATLAB and Simulink. The control designed for the decoupled system has been proved to be suitable for the original non-linear model.

Contents

1 Introduction **5**

1.1 Objectives and Methodology 5

1.2 Background and Related Projects 6

2 Mechanical Model **7**

2.1 Mechanical Model 7

2.2 Wheel Model and Forces 9

2.3 Track Following 11

2.4 Final System 14

3 Linearized Model **15**

3.1 Linearization 15

3.2 Analysis of the Decoupled System 18

4 Design of the Control Algorithms **23**

4.1 Design of the Speed Subsystem 23

4.2 Design of the Distance Controller 27

4.2.1 Proportional Controller 27

4.2.2 Proportional Controller with Feed-forward 31

5 Estimation of the Parameters of the Robot 41

5.1 Geometrical Parameters 41

5.2 Viscous Friction Coefficient 42

5.3 Summary of Parameter Values 42

6 Simulations 45

6.1 Simulations Using the Linearized Model 45

6.2 Simulations Using the Nonlinear Model 50

A Budget 57

A.1 Labour costs 57

A.2 Development tools and office material 58

A.3 Total cost 58

Chapter 1

Introduction

This project has come to be in the context of a bigger project on platooning systems, which aims to obtain interactive vehicles circulating in the same path.

In a previous thesis, a two-wheeled robot was built and programmed to track a line and following projects have expanded and improved that work. However, the model used in them to design the control was kinematic, but not dynamic. It would be hence interesting to study if adding the effects of the forces and torques improves the response of the robot. The present work provides a control algorithm for a more complete mechanical model.

1.1 Objectives and Methodology

The primary objective of this thesis is to redesign the control of the two-wheeled tracking robot for a complete mechanical system. This can be divided in three specific objectives, which are:

a) Create a new model of the two-wheeled robot that includes all of its mechanical behaviour.

b) Design the control for that model.

c) Perform simulations to check the functioning of the designed control.

In order to accomplish these objectives, all the calculations and simulations, as well as the graphics used in the system analysis have been done using MATLAB and Simulink.

1.2 Background and Related Projects

This project is related to various Bachelor's Thesis previously done by other students. The one that has influenced more this work is *Control Design and Implementation for a Line Tracker Vehicle* [1], in which the author designed and built the hardware and firmware of a two-wheeled line tracking robot. The other works conducted within this project are:

- *Disseny i implementació d'un sistema de comunicacions WiFi per a una xarxa de vehicles autònoms* [2], in which the author created a local network in order to remotely control the robots.
- *Design of Controllers and its Implementation for a Line Tracker Vehicle* [3], in which the two thesis above were put together.
- *Hardware Design and Implementation of Two-Wheleed Vehicle Robots for a Platooning System* [4], in which the hardware was optimized and the control improved.
- *Adaptative Cruise Control, an Scaled Model: Platooning Using Two-Wheeled Robots* [5], in which more robots were created in order to program an interconnected platooning position control.
- *Control of a Four In-Wheel Motor Drive Electric Vehicle* [6], in which an optimized control was developed for a four in-wheel motor independent drive vehicle.

Chapter 2

Mechanical Model

The first step to control the two-wheeled robot is to design a mathematical model that defines its movement and line-tracking behaviour. The final control algorithm will let the user define the cruise speed while maintaining the robot over the chosen line.

In order to do that, the mechanics of the robot have been analyzed, treating its body as rigid and assuming that neither the caster nor the driving wheels slip in any direction.

In addition, the curve to be followed has been modelled, as well as its distance to the tracking point, which a desired control would make equal to zero.

2.1 Mechanical Model

Figure 2.1 is a schematic approximation of the real life two-wheeled robot that illustrates and names the elements and distances that are important for the model, as well as the forces considered: a longitudinal one and a traversal one in each of the driving wheels (the forces in the caster are considered negligible). Curve $\sigma(q)$ is a representation of the line to follow.

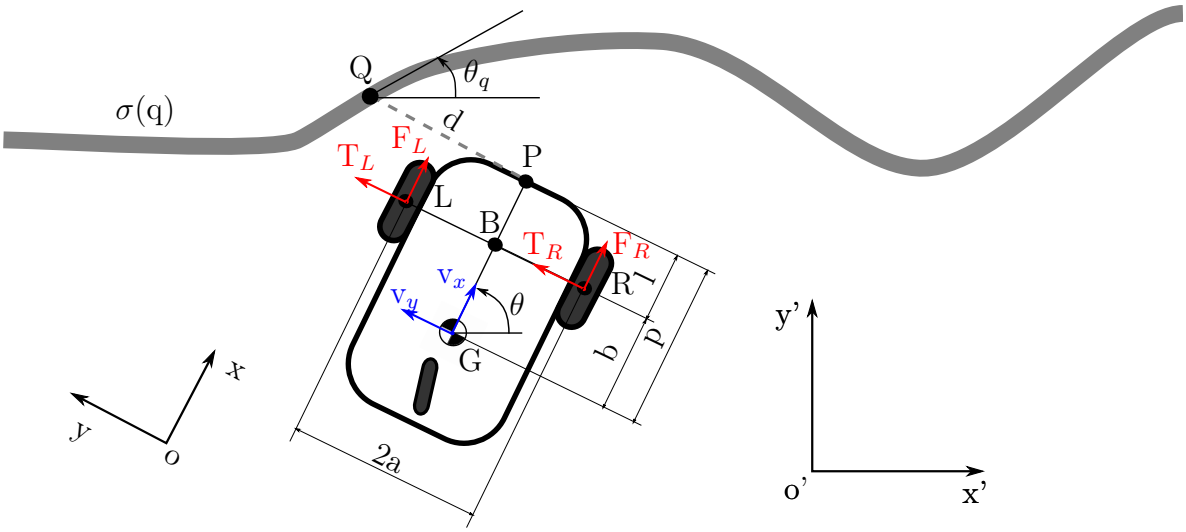

Figure 2.1: Scheme of the robot and the line it follows

Treating the robot as a rigid body, its dynamics can be studied by applying the Conservation of Momentum and the Conservation of Angular Momentum [7].

The velocities and accelerations have been calculated in a general reference frame fixed on the floor (x',y'), but the vectors are expressed in moving basis x-y, which is fixed to the robot and has its angular velocity. Both basis are shown in Figure 2.1

Applying the Conservation of Momentum the following equations can be found

$$\begin{aligned} F_R + F_L &= ma_x \\ T_R + T_L &= ma_y, \end{aligned} \tag{2.1}$$

where a_x and a_y are the respective components of the acceleration vector $\boldsymbol{a} = (a_x, a_y, 0)^\top$, which is the derivative of the velocity vector $\boldsymbol{v} = (v_x, v_y, 0)^\top$. It is to note that, as the vectors are expressed in a moving basis, the derivation must take the basis rotation into account (Equation (2.2)). [7]

$$\begin{pmatrix} a_x \\ a_y \\ 0 \end{pmatrix} = \dot{\boldsymbol{v}} = \begin{pmatrix} \dot{v}_x \\ \dot{v}_y \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \dot{\theta} \end{pmatrix} \wedge \begin{pmatrix} v_x \\ v_y \\ 0 \end{pmatrix} = \begin{pmatrix} \dot{v}_x - \dot{\theta} v_y \\ \dot{v}_y + \dot{\theta} v_x \\ 0 \end{pmatrix} \tag{2.2}$$

Naming the angular velocity as

$$\dot{\theta} = \omega \tag{2.3}$$

and substituting according to (2.2), Equations (2.1) can be expressed as

$$\begin{aligned} m(\dot{v}_x - \omega v_y) &= (F_R + F_L) \\ m(\dot{v}_y + \omega v_x) &= (T_R + T_L) \end{aligned} \tag{2.4}$$

By applying the Theorem of Conservation of Angular Momentum to the Center of Mass, the following equation can be found

$$b(T_R + T_L) + a(F_R - F_L) = I_z\ddot{\theta} \tag{2.5}$$

where I_z is the moment of inertia of the robot in the vertical axis. [7]

According to (2.3), this equation can also be expressed as:

$$I_z\dot{\omega} = b(T_R + T_L) + a(F_R - F_L) \tag{2.6}$$

2.2 Wheel Model and Forces

To find the values of the forces, a wheel model must be chosen. In *Vehicle Dynamics and Control* [8], the author suggests a model that includes the possible longitudinal and traversal slips of the driving wheels and defines their relation to the forces in the wheels. If this model were used, the longitudinal and rotational velocity of the wheels would have to be measured in order to obtain the slip ratio, which influences the forces. As that would further complicate the models, the slip is considered negligible.

Since no slip is considered, the only longitudinal forces that appear in the model are the ones produced by the electric motor that controls each wheel.

Under the assumption that the two wheels and their engines are equal, the dynamics of the wheel-motor assembly can be described as

$$\begin{aligned} I\dot{\omega}_R + B_f\omega_R &= \tau_R - F_R r \\ I\dot{\omega}_L + B_f\omega_L &= \tau_L - F_L r \end{aligned} \tag{2.7}$$

where I is the moment of inertia of the wheel in the direction of the axis, B_f is the total viscous friction coefficient between the rotor and the wheel, ω_R and ω_L are the angular

velocities of the wheels, τ_R and τ_L are the torques applied to them and r is the radius of the tires. [9]

It should also be noted that the applied torque is directly proportional to the current given to the motor, which is interesting for the experimental application of the designed control algorithm to the robot.

$$\begin{aligned} \tau_R &= Ki_R \\ \tau_L &= Ki_L \end{aligned} \tag{2.8}$$

If the moment of inertia of the wheels is negligible, the longitudinal forces of each wheel are:

$$\begin{aligned} F_R &= \frac{\tau_R}{r} - \frac{B_f}{r}\omega_R \\ F_L &= \frac{\tau_L}{r} - \frac{B_f}{r}\omega_L \end{aligned} \tag{2.9}$$

Since no slip is considered, the angular velocity of each wheel can be calculated in terms of the velocity of the center of the wheel in the x direction, i.e.:

$$\begin{aligned} \omega_R &= \frac{v_{R,x}}{r} \\ \omega_L &= \frac{v_{L,x}}{r} \end{aligned} \tag{2.10}$$

As the robot is considered a rigid body, the velocities in the wheels are correlated with the velocity in the center of mass and the angular velocity of the robot. [7]

$$\boldsymbol{v_R} = \begin{pmatrix} v_{R,x} \\ v_{R,y} \\ 0 \end{pmatrix} = \begin{pmatrix} v_x + \omega a \\ v_y + \omega b \\ 0 \end{pmatrix} \qquad \boldsymbol{v_L} = \begin{pmatrix} v_{L,x} \\ v_{L,y} \\ 0 \end{pmatrix} = \begin{pmatrix} v_x - \omega a \\ v_y + \omega b \\ 0 \end{pmatrix} \tag{2.11}$$

Substituting in (2.9) according to (2.10) and (2.11), the longitudinal forces of the robot can be expressed as

$$\begin{aligned} F_R &= \frac{\tau_R}{r} - \frac{B_f}{r^2}(v_x + \omega a) \\ F_L &= \frac{\tau_L}{r} - \frac{B_f}{r^2}(v_x - \omega a) \end{aligned} \tag{2.12}$$

and the equations found in (2.4) and (2.6) are

$$\begin{aligned} m(\dot{v}_x - \omega v_y) &= \frac{1}{r}(\tau_R + \tau_L) - 2\frac{B_f}{r^2}v_x \\ m(\dot{v}_y + \omega v_x) &= (T_R + T_L) \\ I_z\dot{\omega} &= b(T_R + T_L) + \frac{a}{r}(\tau_R - \tau_L) - 2\frac{B_f a^2}{r^2}\omega \end{aligned} \tag{2.13}$$

T_R and T_L are the forces that prevent the wheels to move in the y axis, but they have unknown values, so another equation is needed in order to have a defined system. Going back to the velocities in the wheels found in (2.11), it can be seen that the velocity in the y axis is not zero, as it should be under the assumption that there is no lateral slip in the wheels. Making that value equal to zero provides another equation.

$$v_{R,y} = v_{L,y} = v_y + \omega b = 0 \tag{2.14}$$

By derivation, this can be expressed as

$$\dot{v}_y = -\dot{\omega}b \tag{2.15}$$

Applying this newly found relation to system (2.13), the following equations can be found.

$$\begin{aligned} \dot{v}_x &= \frac{1}{mr}(\tau_R + \tau_L) - 2\frac{B_f}{mr^2}v_x - b\omega^2 \\ \dot{\omega} &= \frac{1}{I_z + mb^2}\left(\frac{a}{r}(\tau_R - \tau_L) - 2\frac{B_f a^2}{r^2}\omega + mb\omega v_x\right) \end{aligned} \tag{2.16}$$

2.3 Track Following

In order to complete the model, the robot must be situated in space and in reference to the chosen track to follow, i.e., the curve noted as $\sigma(q)$. The following development shows the procedure used to find an equation for d, one of the two parameters to control[1].

As shown in Figure 2.1, parameter d notes the distance between the tracking point (P) and the point being tracked (Q). Equation (2.17) shows the situation of point

Q, which can be defined by its components in the x'-y' reference system (see Figure 2.1), as well as by its belonging to $\sigma(q)$.

$$\begin{pmatrix} x'_Q \\ y'_Q \end{pmatrix} = \begin{pmatrix} \sigma_{x'}(q) \\ \sigma_{y'}(q) \end{pmatrix} \tag{2.17}$$

Using a rotation matrix and and known shape measurements of the car, point Q can be situated in reference to the center of masses as follows

$$\begin{pmatrix} x'_Q \\ y'_Q \end{pmatrix} = \begin{pmatrix} x'_G \\ y'_G \end{pmatrix} + R(\theta) \begin{pmatrix} p \\ d \end{pmatrix}, \tag{2.18}$$

where the mentioned matrix is

$$R(\theta) = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}. \tag{2.19}$$

By derivation from Equation (2.18), the following expression is obtained:

$$\begin{pmatrix} \dot{x}'_G \\ \dot{y}'_G \end{pmatrix} + \frac{d}{dt} R(\theta) \begin{pmatrix} p \\ d \end{pmatrix} + R(\theta) \begin{pmatrix} 0 \\ \dot{d} \end{pmatrix} = \dot{q} \begin{pmatrix} \frac{\partial}{\partial q}\sigma_{x'}(q) \\ \frac{\partial}{\partial q}\sigma_{y'}(q) \end{pmatrix} \tag{2.20}$$

As x'-y' is a fix reference,

$$\begin{aligned} \dot{x'}_G &= v_{x'} \\ \dot{y'}_G &= v_{y'} \\ \dot{\theta} &= \omega, \end{aligned} \tag{2.21}$$

Equation (2.20) can be expressed as

$$\begin{pmatrix} v_{x'} \\ v_{y'} \end{pmatrix} + \frac{d}{dt} R(\theta) \begin{pmatrix} p \\ d \end{pmatrix} + R(\theta) \begin{pmatrix} 0 \\ \dot{d} \end{pmatrix} = \dot{q} \begin{pmatrix} \cos\theta_q \\ \sin\theta_q \end{pmatrix} \tag{2.22}$$

and $\dot{d}$ and $\dot{q}$ can be isolated from (2.22) order to obtain Equations (2.27).

$$\begin{pmatrix} 0 \\ \dot{d} \end{pmatrix} = R^{-1}(\theta) \left[- \begin{pmatrix} v_{x'} \\ v_{y'} \end{pmatrix} - \frac{d}{dt} R(\theta) \begin{pmatrix} p \\ d \end{pmatrix} + \dot{q} \begin{pmatrix} \cos\theta_q \\ \sin\theta_q \end{pmatrix} \right] \tag{2.23}$$

$$\begin{pmatrix} 0 \\ \dot{d} \end{pmatrix} = -R^{-1}(\theta) \begin{pmatrix} v_{x'} \\ v_{y'} \end{pmatrix} - \omega R^{-1}(\theta) \frac{\partial R(\theta)}{\partial \theta} \begin{pmatrix} p \\ d \end{pmatrix} + R^{-1}(\theta)\dot{q} \begin{pmatrix} \cos\theta_q \\ \sin\theta_q \end{pmatrix} \tag{2.24}$$

$$R^{-1}(\theta) = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \quad , \quad \frac{\partial R(\theta)}{\partial \theta} = \begin{pmatrix} -\sin\theta & -\cos\theta \\ \cos\theta & -\sin\theta \end{pmatrix} \tag{2.25}$$

$$R^{-1}(\theta)\frac{\partial R(\theta)}{\partial \theta} = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \tag{2.26}$$

$$\begin{aligned} \dot{q} &= \frac{v_{x'}\cos\theta + v_y \sin\theta - \omega d}{\cos\theta\cos\theta_q + \sin\theta\sin\theta_q} \\ \dot{d} &= v_{x'}\sin\theta - v_{y'}\cos\theta - \omega p + \dot{q}(-\sin\theta\cos\theta_q + cos\theta\sin\theta_q) \end{aligned} \tag{2.27}$$

This can be further simplified by applying the following trigonometrical equations

$$\begin{aligned} \sin(\theta - \theta_q) &= \sin\theta\cos\theta_q - cos\theta\sin\theta_q \\ \cos(\theta - \theta_q) &= \cos\theta\cos\theta_q + \sin\theta\sin\theta_q \end{aligned} \tag{2.28}$$

and defining a new angle:

$$\theta_e = \theta - \theta_q \tag{2.29}$$

The resulting equations are as follows

$$\begin{aligned} \dot{q} &= \frac{v_{x'}\cos\theta + v_{y'}\sin\theta - \omega d}{\cos\theta_e} \\ \dot{d} &= v_{x'}\sin\theta - v_{y'}\cos\theta - \omega p + \dot{q}\sin\theta_e \end{aligned} \tag{2.30}$$

and $\dot{d}$ can be further simplified by substituting $\dot{q}$ by its equation.

$$\dot{d} = v_{x'}\sin\theta - v_{y'}\cos\theta - \omega p - \tan\theta_e(v_{x'}\cos\theta + v_{y'}\sin\theta - \omega d) \tag{2.31}$$

As Equations (2.30) are expressed in the x'-y' axis, rotation matrix R(θ) is used to find the expressions of $\dot{d}$ and $\dot{q}$ in the x-y basis so they are expressed with the same parameters as in Equations (2.16).

$$\begin{pmatrix} v_{x'} \\ v_{y'} \end{pmatrix} = R^{-1}(\theta) \begin{pmatrix} v_x \\ v_y \end{pmatrix} \tag{2.32}$$

Using these variables, the equations are much more short and simple.

$$\begin{aligned} \dot{d} &= -v_y - \omega p - \tan\theta_e(v_x - \omega d) \\ \dot{q} &= \frac{v_x - \omega d}{\cos\theta_e} \end{aligned} \tag{2.33}$$

According to the relation between v_y and ω seen in(2.14) and using the following parameter relation (as seen in Figure 2.1) $\dot{d}$ can be further simplified.

$$l = p - b \tag{2.34}$$

$$\dot{d} = -l\omega - \tan\theta_e(v_x - \omega d) \tag{2.35}$$

2.4 Final System

In order to define the system that will be studied in the following chapters, the study variables must be specified. The chosen variables are $\dot{d}$, $\dot{\theta}_e$, $\dot{v}_x$ and $\dot{\omega}$.

The equation for $\dot{\theta}_e$ can be found by derivation from its definition (Equation (2.29)).

$$\dot{\theta}_e = \dot{\theta} - \dot{\theta}_q \tag{2.36}$$

where $\dot{\theta}$ is ω as defined in the previous sections and

$$\dot{\theta}_q = c(q)\dot{q}. \tag{2.37}$$

$\dot{q}$ is as defined in Equation (2.33) and $c(q)$ is the curvature of the track for every q.

$$c(q) = \frac{\partial\theta_q}{\partial q} \tag{2.38}$$

The final four equations of the model are:

$$\begin{aligned} \dot{d} &= -l\omega - \tan\theta_e(v_x - \omega d) \\ \dot{\theta}_e &= \omega - c(q)\frac{v_x - \omega d}{\cos\theta_e} \\ \dot{v}_x &= \frac{1}{mr}(\tau_R + \tau_L) - 2\frac{B_f}{mr^2}v_x - b\omega^2 \\ \dot{\omega} &= \frac{1}{I_z + mb^2}\left(\frac{a}{r}(\tau_R - \tau_L) - 2\frac{B_f a^2}{r^2}\omega + mb\omega v_x\right) \end{aligned} \tag{2.39}$$

Chapter 3

Linearized Model

As the suggested system is nonlinear, it must be linearized in order to apply control theory for linear systems.

The state space of the system around an equilibrium point is found in this chapter, as well as the decoupled transfer functions for an idealized case. This will be used in following simulations and to design an adequate controller.

3.1 Linearization

The system obtained in the previous chapter (Equations 2.39) is a nonlinear system of the form

$$\dot{z} = f(z) + g(z)u, \tag{3.1}$$

where state variables (z) are d, θ_e, v_x and ω and the control signals (u) are τ_R and τ_L. If the system operates around an equilibrium point and the signals involved are small, it can be approximated by a linear system [10]. The linearization is based on the expansion of the nonlinear function into a Taylor series around the operating point [10] and the resulting linear time-invariant model takes the form

$$\begin{aligned} \dot{\boldsymbol{z}} &= A\boldsymbol{z} + B\boldsymbol{u} \\ \boldsymbol{y} &= C\boldsymbol{z} + D\boldsymbol{u}, \end{aligned} \tag{3.2}$$

where

$$\boldsymbol{z} = \begin{pmatrix} d \\ \theta_e \\ v_x \\ \omega \end{pmatrix} \tag{3.3}$$

$$\boldsymbol{u} = \begin{pmatrix} \tau_R \\ \tau_L \end{pmatrix} \tag{3.4}$$

$$\boldsymbol{y} = \begin{pmatrix} d \\ v_x \end{pmatrix} \tag{3.5}$$

and

$$A_{ij} = \frac{\partial \dot{z}_i}{\partial z_j}\bigg]_{z_0,u_0} \quad ; \quad B_{ij} = \frac{\partial \dot{z}_i}{\partial u_j}\bigg]_{z_0,u_0} \tag{3.6}$$

To find the equilibrium values z_0 and u_0, System (2.39) is solved with $\dot{z} = 0$. The desired equilibrium conditions are defined as

$$\begin{aligned} d_0 &= 0 \\ v_{x,0} &= v_d \end{aligned} \tag{3.7}$$

and, as a simplification, the curvature is assumed to be constant.

$$c(q) = c \tag{3.8}$$

The equilibrium values found are:

$$\begin{aligned} \theta_{e,0} &= \arcsin\left(-lc\right) \\ \omega_0 &= \frac{c}{\alpha} v_d \\ \tau_{R,0} &= \frac{bmrc}{2\alpha}\left(\frac{c}{\alpha} - \frac{1}{a}\right) v_d^2 + \frac{B_f}{r}\left(1 + \frac{ca}{\alpha}\right) v_d \\ \tau_{L,0} &= \frac{bmrc}{2\alpha}\left(\frac{c}{\alpha} + \frac{1}{a}\right) v_d^2 + \frac{B_f}{r}\left(1 - \frac{ca}{\alpha}\right) v_d, \end{aligned} \tag{3.9}$$

where

$$\alpha = \sqrt{1 - l^2 c^2} \tag{3.10}$$

Finally, the matrices of the system are as follows.

$$A = \begin{pmatrix} -l\frac{c^2}{\alpha^2}v_d & -\frac{1}{\alpha^2}v_d & l\frac{c}{\alpha} & -l \\ \frac{c^2}{\alpha^2}v_d & l\frac{c^2}{\alpha^2}v_d & -\frac{c}{\alpha} & 1 \\ 0 & 0 & -\frac{2B_f}{mr^2} & -2b\frac{c}{\alpha}v_d \\ 0 & 0 & \frac{mb}{I_z+mb^2}\frac{c}{\alpha}v_d & \frac{1}{I_z+mb^2}\left(mbv_d - \frac{2B_f a^2}{r^2}\right) \end{pmatrix} \tag{3.11}$$

$$B = \frac{1}{r}\begin{pmatrix} 0 & 0 \\ 0 & 0 \\ \frac{1}{m} & \frac{1}{m} \\ \frac{a}{I_z+mb^2} & -\frac{a}{I_z+mb^2} \end{pmatrix} \tag{3.12}$$

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{3.13}$$

$$D = 0 \tag{3.14}$$

For $c = 0$ the system described in (3.11) and (3.12) becomes

$$A = \begin{pmatrix} 0 & -v_d & 0 & -l \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -2\frac{B_f}{mr^2} & 0 \\ 0 & 0 & 0 & -\frac{1}{I_z+mb^2}\left(2\frac{B_f a^2}{r^2} - mbv_d\right) \end{pmatrix} \tag{3.15}$$

$$B = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ \frac{1}{mr} & \frac{1}{mr} \\ \frac{a}{r(I_z+mb^2)} & -\frac{a}{r(I_z+mb^2)} \end{pmatrix} \tag{3.16}$$

Defining

$$u_d = (\tau_L - \tau_R) \tag{3.17}$$

$$u_v = (\tau_L + \tau_R) \tag{3.18}$$

or, equivalently

$$\boldsymbol{u_1} = T\boldsymbol{\tau} = \begin{pmatrix} -1 & 1 \\ 1 & 1 \end{pmatrix} \boldsymbol{\tau} \tag{3.19}$$

the system can be rewritten as

$$\begin{aligned} \dot{\boldsymbol{z}} &= A\boldsymbol{z} + B'\boldsymbol{u_1} \\ \boldsymbol{y} &= C\boldsymbol{z} + D\boldsymbol{u_1}, \end{aligned} \tag{3.20}$$

where $\boldsymbol{u_1} = (u_d, u_v)^\top$ and

$$B' = BT^{-1} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & \frac{1}{mr} \\ -\frac{a}{r(I_z+mb^2)} & 0 \end{pmatrix} \tag{3.21}$$

Then, the system is decoupled, and the transfer functions that relate u_d and u_v with d and v_x are

$$\begin{aligned} G_d(s) &= \frac{D(s)}{U_d(s)} = \frac{ar(ls + v_d)}{s^2(r^2(I_z + mb^2)s + 2B_f a^2 - mv_d br^2)} \\ G_v(s) &= \frac{V_x(s)}{U_v(s)} = \frac{r}{mr^2 s + 2B_f} \end{aligned} \tag{3.22}$$

It is clear that the speed can be controlled through u_2 with a simple PI controller, that will guarantee asymptotic stability at v_d. However, transfer function $G_d(s)$ reveals that the control of d is more complicated.

3.2 Analysis of the Decoupled System

As found in the previous chapter, transfer functions (3.22) are the ones used in the case of tracking a straight line with the linear model.

$G_v(s)$ has no zeros and one pole that does not depend on the chosen velocity. As the only pole is negative, the open loop system is stable [10]. That pole is:

$$p_v = -\frac{2B_f}{mr^2} \tag{3.23}$$

and substituting the parameters with its numerical values, found in 5.2:

$$p_v = -0,2144 \tag{3.24}$$

The following plot shows the Root Locus for any v_d, as the poles don't depend on it. The numerical values used are the ones told above. The Figure shows that the system is stable for any proportional controller. Adding an integrator will help reduce the error [10].

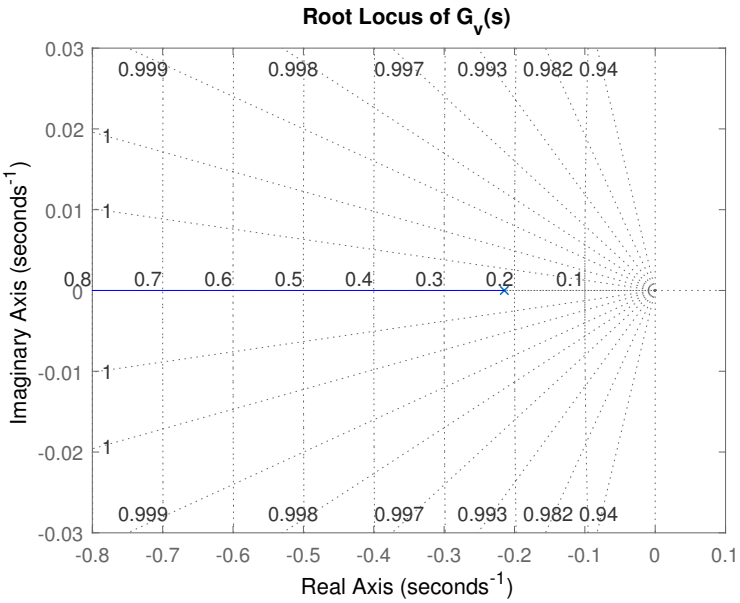

Figure 3.1: Root Locus of $G_d(s)$

$G_d(s)$ has two zeros and three poles. One of the zeros and two of the poles depend on the chosen velocity (v_d). As two of the poles are on the imaginary axis, the open loop system is unstable [10].

$$\begin{aligned} z_d &= -\frac{v_d}{l} \\ p_{d,1;2} &= 0 \\ p_{d,3} &= \frac{mv_d br^2 - 2B_f a^2}{mb^2r^2 + Ir^2} \end{aligned} \tag{3.25}$$

A Root Locus analysis [10] has been done in order to study the behaviour of the system in closed loop for negative feedback. The following plots show the root locus for different values of v_d, using the numerical values of the parameters from Table 5.2

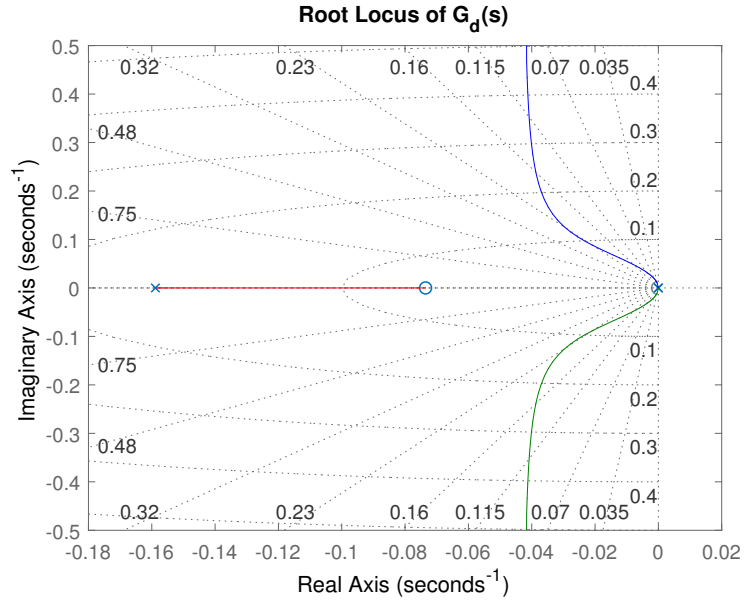

Figure 3.2: Root Locus of $G_d(s)$ for $v_d = 0,005m/s$

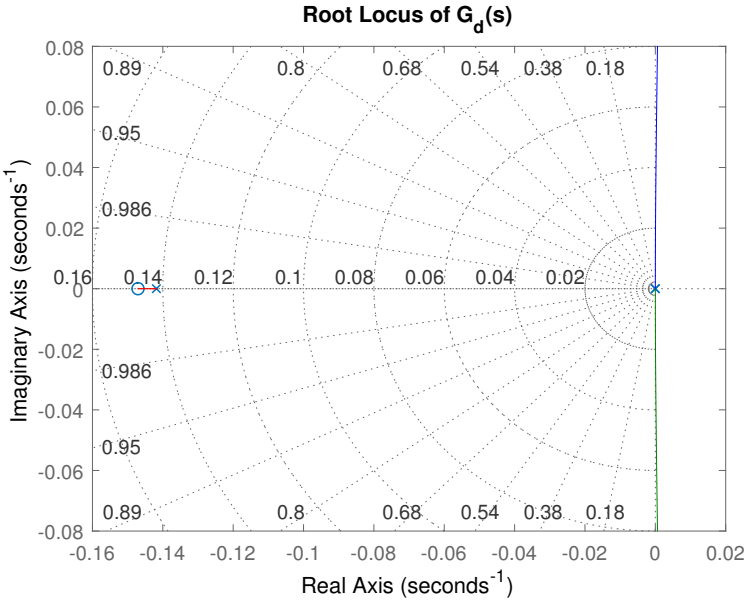

Figure 3.3: Root Locus of $G_d(s)$ for $v_d = 0,01m/s$

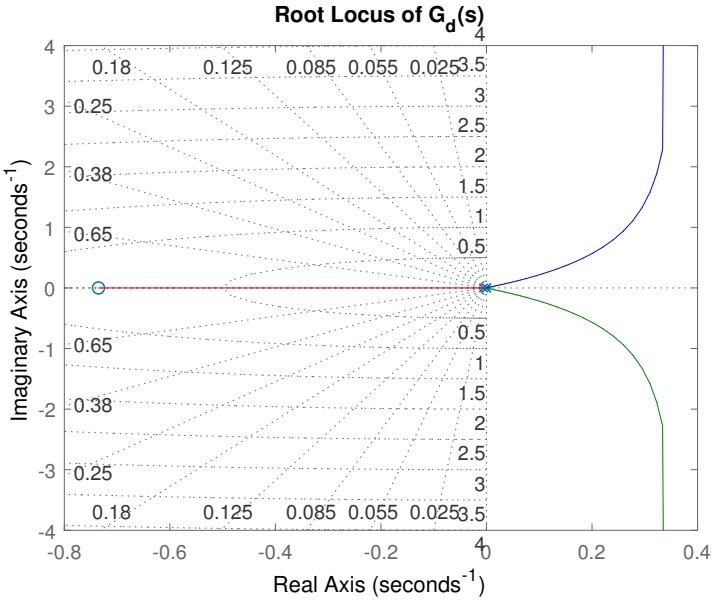

Figure 3.4: Root Locus of $G_d(s)$ for $v_d = 0,05m/s$

Observing the plots, it is clear that the system is just stable for a certain interval of v_d. In order to make the system stable, the centroid of the asymptotes must be negative [10].

The centroid is

$$c = \frac{1}{n-m}(\sum p_i - \sum z_i) \tag{3.26}$$

or, replacing the poles and zeros from (3.25)

$$c = \frac{1}{2}\left(\frac{r^2 mbv_d - 2B_f a^2}{r^2(I_z + mb^2)} + \frac{v_d}{l}\right) \tag{3.27}$$

Imposing $c < 0$

$$\frac{r^2 mbv_d - 2B_f a^2}{r^2(I_z + mb^2)} + \frac{v_d}{l} < 0 \tag{3.28}$$

the critical speed is found,

$$v_d^{\text{crit}} = \frac{2B_f a^2 l}{r^2(I_z + mb^2 + mbl)} \tag{3.29}$$

This value is correct if v_d is positive, i.e. if the robot doesn't go backwards. The other values can be assumed to be always positive: the mass, inertia and the viscous

friction coefficient can't be negative and the distances will always be positive as long as the placement of the wheels and the sensor is as shown in Figure 2.1.

According to the values of the parameters described in Table 5.2, that speed is:

$$v_d^{\text{crit}} = 0,0097 m/s \tag{3.30}$$

As this value is very low, an alternative option to control the robot at higher speeds has been considered, as explained in the following chapter. A possible reason of this low speed is that the value of B_f has been underestimated.

In short, the control of the distance d becomes complicated because the parameters strongly affect the placement of the open loop poles.

Chapter 4

Design of the Control Algorithms

After linearizing the system, the design of the control algorithm has been made for the case of the decoupled system (Equations 3.22).

4.1 Design of the Speed Subsystem

The robot is controlled with a PI controller with the form:

$$C_v(s) = k_{pv} + k_{iv}\frac{1}{s} \tag{4.1}$$

or, defining $z_c = K_{iv}/k_{pv}$,

$$C_v(s) = k_{pv}\frac{s + z_c}{s} \tag{4.2}$$

The resulting closed loop system is then defined by the following transfer function:

$$T_d(s) = \frac{C_v(s)G_v(s)}{1 + C_v(s)G_v(s)}, \tag{4.3}$$

which, replacing according to (4.2) and (3.22), describes a system with two poles and a zero [10].

The desired values of the poles can be found by defining the target dynamics of the system. Two values that can be interesting are the settling time (t_s) defined as the

time till the system is at 98% of its final value, and the maximum overshot performance (M_p) which is the maximum overshot from the desired value for one unit.

If the poles are:

$$s_d = -\sigma_d \pm j\omega_d, \tag{4.4}$$

then t_s and M_p can be described according to them:

$$t_s = \frac{4}{\sigma_d} \tag{4.5}$$

$$M_p = e^{-\frac{\pi\sigma_d}{\omega_d}} \tag{4.6}$$

and the design of the controller can be made by using the root locus approach [10].

If the characteristic equation of the root locus is:

$$1 + C_d(s)G_d(s) = 0 \tag{4.7}$$

or, replacing by (4.2) and (3.22),

$$1 + \frac{k_{pd}}{mr}\frac{s + z_c}{s(s + 2\frac{B_f}{mr^2})} = 0, \tag{4.8}$$

the module and phase conditions obtained from it are:

$$\left|\frac{k_{pd}}{mr}\frac{s + z_c}{s(s + 2\frac{B_f}{mr^2})}\right| = 1 \tag{4.9}$$

$$\angle\left(\frac{k_{pd}}{mr}\frac{s + z_c}{s\left(s + 2\frac{B_f}{mr^2}\right)}\right) = -\pi \tag{4.10}$$

The chosen poles have to keep this conditions, i.e., according to (4.10):

$$\angle(-\sigma_d + j\omega_d + z_c) - \angle(-\sigma_d + j\omega_d) - \angle\left(-\sigma_d + j\omega_d + 2\frac{B_f}{mr^2}\right) = -\pi \tag{4.11}$$

or

$$\arctan\left(\frac{\omega_d}{-\sigma_d + z_c}\right) - \arctan\left(\frac{\omega_d}{-\sigma_d}\right) - \arctan\left(\frac{\omega_d}{-\sigma_d + 2\frac{B_f}{mr^2}}\right) = -\pi, \tag{4.12}$$

and, according to (4.9):

$$\left|\frac{k_{pd}}{mr}\frac{-\sigma_d + j\omega_d + z_c}{-\sigma_d + j\omega_d(-\sigma_d + j\omega_d + 2\frac{B_f}{mr^2})}\right| = 1 \tag{4.13}$$

Finally, the parameters that define the PI controller can be found from the two equations above, obtaining:

$$\begin{aligned} z_c &= \sigma_d + \frac{\omega_d}{\tan\left(\arctan\left(\frac{\omega_d}{-\sigma_d}\right) + \arctan\left(\frac{\omega_d}{-\sigma_d + 2\frac{B_f}{mr^2}}\right) - \pi\right)} \\ k_{pv} &= mr \left| \frac{-\sigma_d + j\omega_d(-\sigma_d + j\omega_d + 2\frac{B_f}{mr^2})}{-\sigma_d + j\omega_d + z_c} \right|, \end{aligned} \tag{4.14}$$

where σ_d and ω_d are defined by Equations (4.5) and (4.6):

$$\begin{aligned} \sigma_d &= \frac{4}{t_s} \\ \omega_d &= -\frac{\pi\sigma_d}{\ln(M_p)} \end{aligned} \tag{4.15}$$

Using the values from Table 5.2, and defining the following target dynamics:

$$\begin{aligned} M_p &= 10^{-7} \\ t_s &= 1s \end{aligned} \tag{4.16}$$

the controller parameters defined by Equations (4.14) and (4.15) are:

$$\begin{aligned} z_c &= 2,1332 \\ k_{pv} &= 0,3202 \end{aligned} \tag{4.17}$$

Figure 4.1 shows the Root Locus for that case, with the closed loop poles of the system for the designed k_{pv} in black. The unit step response for the system is in Figure 4.2.

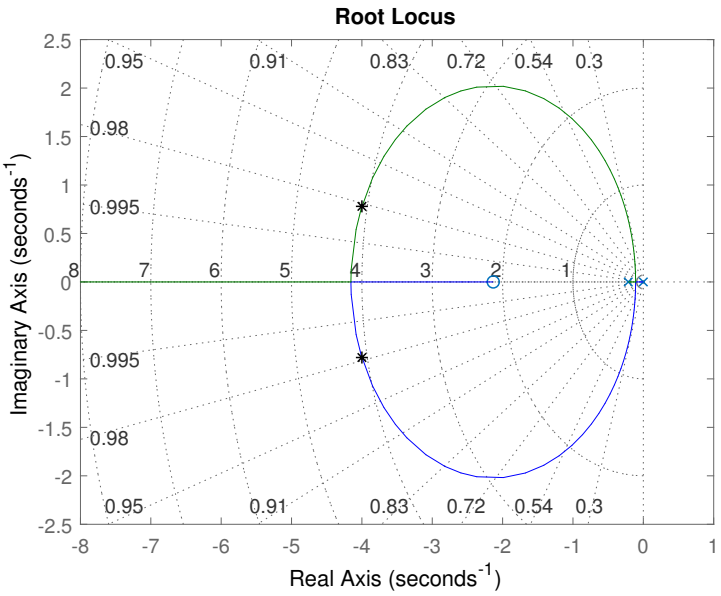

Figure 4.1: Root locus of the velocity system with the designed PI controller

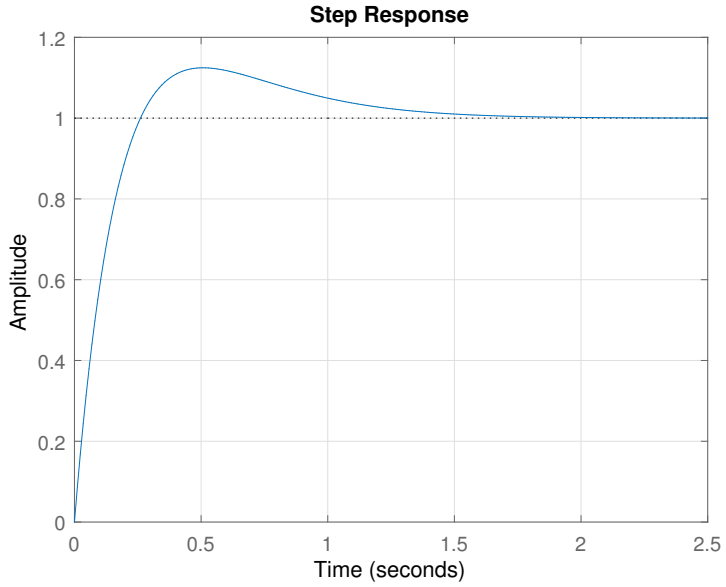

Figure 4.2: Unit step response of the velocity system with the designed PI controller

Looking at the step response, it is obvious that the maximum overshot performance is not as defined, but much bigger. This is because the equation used to define M_p is for a system without zeros, and $T_d(s)$ has one zero (z_c), which makes the overshot

bigger. Expecting this fact, the theoretical value of M_p has been given a small value (10^{-7}).

4.2 Design of the Distance Controller

With the linearization of the system, the desired value of the velocity (v_d) has become a system parameter. This parameter does not appear in the velocity subsystem, but it critically affects the distance control, as seen in Section 3.2. For that reason, the following two options are considered:

a) Assuming that the value of the desired longitudinal speed is sufficiently small to obtain the desired performance, with a proportional controller.

b) For higher speeds, damping the angular speed, which will require its measurement or a state observer in case this magnitude is not available.

4.2.1 Proportional Controller

For option a), as the transfer function has two pure integrators, the stationary error will always be zero. A proportional controller can be used to give the system the desired settling time.

$$u_1 = -k_{pd}(d - d_{\text{ref}}) \tag{4.18}$$

If the system has two double closed loop poles with the form $s_d = -\sigma_d \pm j\omega_d$, t_s and M_p can be defined as in (4.5) and (4.6) and the desired value of k_{pd} can be found using the characteristic equation of the root locus, in a similar way as done in Section 4.1.

Of course, this calculation only works as long as those double poles dominate the third and single pole, i.e.,

$$\frac{\sigma_3}{\sigma_d} \geq 5, \tag{4.19}$$

where σ_3 is the absolute value of the third pole.

As can be seen in the root locus (see Figure 3.2), there is a certain minimum value that t_s can achieve, because the real part of the double loop poles have a maximum absolute value defined by the asymptote. According to (4.5),

$$t_{smin} = \frac{4}{|\sigma|_{max}} = \frac{4}{|c|}, \tag{4.20}$$

where c is the centroid of the asymptote as described in (3.27).

As the value of the centroid becomes more negative with the decreasing of the velocity (see the root locus on Section 3.2), there is a maximum value of the velocity that permits to obtain a certain t_s.

By isolating the velocity in (4.22):

$$v_{dmax} = \frac{2l(t_s B_f a^2 - 4r^2(I_z + mb^2))}{t_s r^2(I_z + mb(l+b))} \tag{4.21}$$

As v_d has to be positive, t_s is restricted.

$$t_s > \frac{4r^2(I_z + mb^2)}{B_f a^2} \tag{4.22}$$

Substituting the parameters by their numerical values from Table 5.2,

$$t_s > 45,485s, \tag{4.23}$$

which means that t_s can't be reduced from 45,485s for any positive velocity. This value seems unacceptable and option b) will likely be a much better choice for the system.

As explained in Section 3.2, it is likely that the value of B_f has been underestimated, as the experiments done in the laboratory show that the robot doesn't have the severe restrictions in velocity and settling time that are encountered here. The following calculations have been done for a value of $B_f = 0,0102$, which allows the system to get a minimum t_s of 1 second with a maximum v_d of 0,2 m/s according to (4.21).

To simplify the following calculations, $G_d(s)$ is written as:

$$G_d(s) = k_d \frac{s - z_d}{s^2(s - p_d)}, \tag{4.24}$$

where

$$\begin{aligned} k_d &= \frac{al}{r(I_z + mb^2)} \\ z_d &= -\frac{v_d}{l} \\ p_d &= \frac{mv_d br^2 - 2B_f a^2}{r^2(I_z + mb^2)} \end{aligned} \tag{4.25}$$

In order to design the proportional controller, which is

$$C_d(s) = k_{pd}, \tag{4.26}$$

the procedure is very similar to the one for the speed controller (Section 4.1), but in this case, only the magnitude condition is needed, as there is only one parameter to find (k_{pd}). This condition is

$$\left| k_{pd} k_d \frac{s - z_d}{s^2(s - p_d)} \right| = 1 \tag{4.27}$$

and, for the desired poles,

$$\left| k_{pd} k_d \frac{-\sigma_d + j\omega_d - z_d}{(-\sigma_d + j\omega_d)^2(-\sigma_d + j\omega_d - p_d)} \right| = 1 \tag{4.28}$$

Finally, k_{pd} is found with

$$k_{pd} = \frac{1}{k_d} \left| \frac{(-\sigma_d + j\omega_d)^2(-\sigma_d + j\omega_d - p_d)}{-\sigma_d + j\omega_d - z_d} \right|, \tag{4.29}$$

where the components of the poles are defined by Equations (4.15). This design is not as easy as the one for the velocity, because for certain values, the designed poles will not be dominant and the system will not respond as desired.

One of the possible solutions is the following. For

$$\begin{aligned} v_d &= 0,12 m/s \\ t_s &= 3s \\ M_p &= 0,01, \end{aligned} \tag{4.30}$$

the gain of the controller is

$$k_{pd} = 1,8897 \tag{4.31}$$

and the double poles are dominant with

$$\frac{\sigma_3}{\sigma_d} = 5,8897 \tag{4.32}$$

This dominance can be appreciated in the following root locus, with the closed loop poles in black.

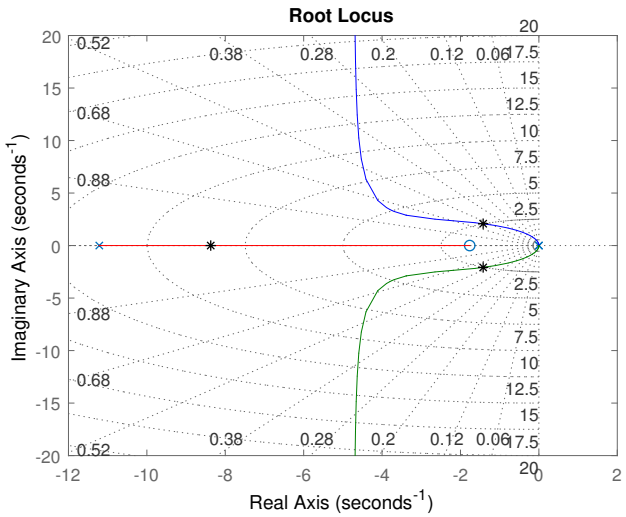

Figure 4.3: Root Locus of the distance system without feed-forward for the designed control

As seen in the velocity control and for the same reason, the desired value of M_p is not the same as the one obtained in the simulation, as seen in Figure 4.4.

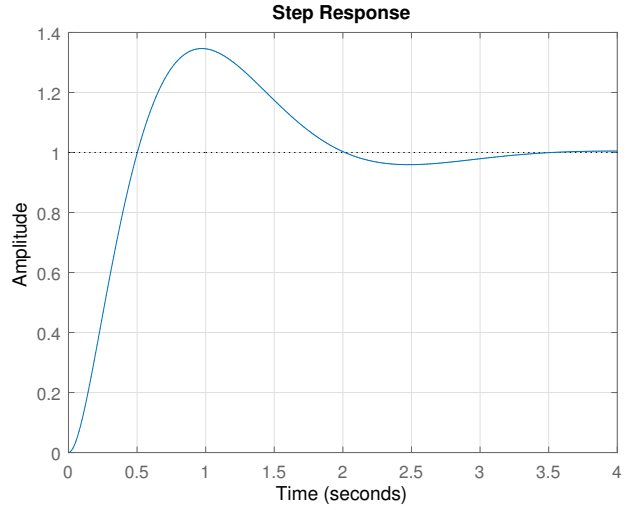

Figure 4.4: Unit step response of the distance system without feed-forward for the designed control

4.2.2 Proportional Controller with Feed-forward

Alternatively to (3.17), for high speeds, the following control variable is proposed

$$u'_d = u_d - \frac{rmbv_d}{a}\omega - k_\omega\omega \tag{4.33}$$

Then, replacing (4.33) in (3.20),

$$\dot{\boldsymbol{z}} = A'\boldsymbol{z} + B'\boldsymbol{u_2} \tag{4.34}$$

where $\boldsymbol{u_2} = (u'_d, u_v)$ and

$$A' = \begin{pmatrix} 0 & -v_d & 0 & -l \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -2\frac{B_f}{mr^2} & 0 \\ 0 & 0 & 0 & -\frac{a}{r(I_z+mb^2)}\left(\frac{2B_f a}{r} + K_\omega\right) \end{pmatrix} \tag{4.35}$$

where ω is assumed to be measured.

Again, the system is decoupled, and the transfer function that relates u_2 with v_x is the same than before (Equations (3.22)), and the one relating u'_d with d is now

$$G'_d(s) = \frac{d(s)}{u'_d(s)} = \frac{ar(ls + v_d)}{s^2\left(r^2(I_z + mb^2)s + 2B_f a^2 + ark_\omega\right)} \tag{4.36}$$

Following the same analysis previously done in Section 3.2

$$c = \frac{1}{2}\left(\frac{-2B_f a^2 - ark_\omega}{r^2(I_z + mb^2)} + \frac{v_d}{l}\right) < 0 \tag{4.37}$$

we get the critical speed, v_d^{crit},

$$v_d^{\text{crit}} = \frac{al(2B_f a + rk_\omega)}{r^2(I_z + mb^2)} \tag{4.38}$$

in which the critical velocity can be as desired by choosing the adequate k_ω.

Another way to see it is that for a chosen v_d, there will be a minimum value of k_ω that stabilizes the system. From (4.37):

$$k_\omega^{crit} = \frac{r^2(I_z + mb^2)v_d - 2B_f la^2}{arl} \tag{4.39}$$

Assuming that the resulting system will be controlled by a proportional controller with the feed forward term, i.e.,

$$u'_d = -k'_{pd}(d - d_{\text{ref}}) \tag{4.40}$$

the system with feed-forward will perform better, as shown in Figure 4.5.

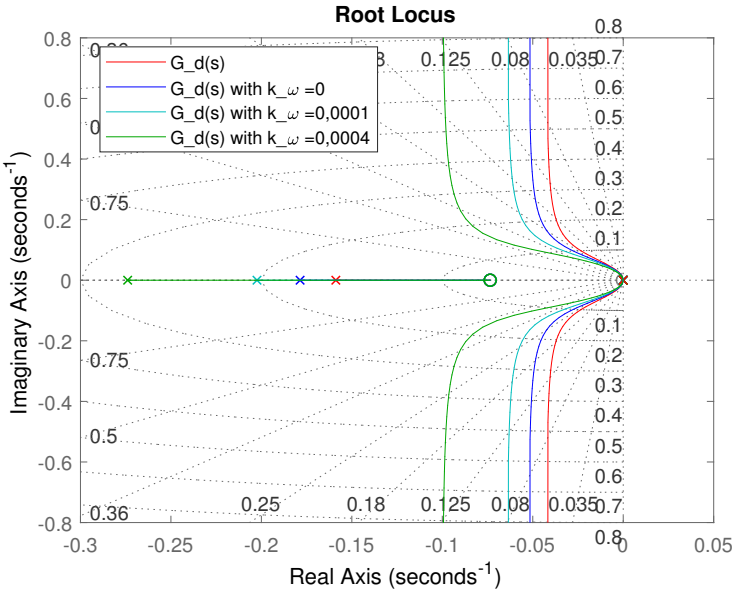

Figure 4.5: Root Locus comparsion between the two control strategies

As the shape of the Root Locus is very similar to the previous system, there is also a minimum t_s that can be achieved for a certain v_d and, in this case, a certain k_w. Following the same steps as in 4.2.1, the maximum velocity for a chosen t_s is found.

$$v_{dmax} = \frac{al(2B_f a + rk_\omega)}{r^2(I_z + mb^2)} - \frac{8l}{t_s} \tag{4.41}$$

By observing the previous formula, it is clear that the maximum velocity for a chosen t_s can be increased by increasing k_ω. Contrary to option a), there is no restriction in the t_s that can be achieved, and it can be as small as wanted.

The minimum k_ω for a chosen t_s and v_d is then found by isolating that parameter in (4.41).

$$k_{\omega min} = \frac{(8l + v_d t_s)r^2(I_z + mb^2) - 2lt_s B_f a^2}{arlt_s} \tag{4.42}$$

Mirroring the simplification done in 4.2.1, $G'_d(s)$ is simplified as:

$$G'_d(s) = k'_d \frac{s + z'_d}{s^2(s - p'_d + p_\omega k_\omega)}, \tag{4.43}$$

where

$$\begin{aligned} k'_d &= \frac{al}{r(I_z + mb^2)} \\ z'_d &= -\frac{v_d}{l} \\ p'_d &= -\frac{2B_f a^2}{r^2(I_z + mb^2)} \\ p_\omega &= \frac{a}{r(I_z + mb^2)} \end{aligned} \tag{4.44}$$

and the proportional controller is

$$C_d(s) = k_{pd}, \tag{4.45}$$

In this case, there are two parameters to be found, so the procedure is very similar to the one for the speed controller (Section 4.1).

The phase and magnitude conditions are:

$$\left| k_{pd} k'_d \frac{s - z'_d}{s^2(s - p'_d + p_\omega k_\omega)} \right| = 1 \tag{4.46}$$

$$\angle \left(k_{pd} k'_d \frac{s - z'_d}{s^2(s - p'_d + p_\omega k_\omega)} \right) = -\pi \tag{4.47}$$

and

$$\arctan\left(\frac{\omega_d}{-\sigma_d - z'_d}\right) - 2\arctan\left(\frac{\omega_d}{-\sigma_d}\right) - \arctan\left(\frac{\omega_d}{-\sigma_d - p'_d + p_\omega k_\omega}\right) = -\pi, \tag{4.48}$$

$$\left| k_{pd} \frac{-\sigma_d + j\omega_d - z'_d}{(-\sigma_d + j\omega_d)^2(-\sigma_d + j\omega_d - p'_d + p_\omega k_\omega)} \right| = 1 \tag{4.49}$$

Finally, the two control parameters can be found with:

$$k_\omega = \frac{1}{p_\omega} \left(\sigma_d + p'_d - \frac{\omega_d}{\tan\left(2\arctan\left(\frac{\omega_d}{-\sigma_d}\right) - \arctan\left(\frac{\omega_d}{-\sigma_d - z'_d}\right) - \pi\right)} \right) \tag{4.50}$$

$$k_{pd} = \frac{1}{k'_d} \left| \frac{(-\sigma_d + j\omega_d)^2(-\sigma_d + j\omega_d - p'_d + p_\omega k_\omega)}{-\sigma_d + j\omega_d - z'_d} \right|, \tag{4.51}$$

where σ_d and ω_d can be found with (4.15).

Of course, this solution only works as long as the placed poles dominate the third one and the resulting system is stable. It is also important to see that this equations depend on the value of v_d (as $z_d = -v_d/l$), so the two control parameters have to be recalculated if the velocity is changed.

For the values of the robot parameters found in Table 5.2 and

$$\begin{aligned} v_d &= 0,12m/s \\ t_s &= 1,1s \\ M_p &= 0,35, \end{aligned} \tag{4.52}$$

the values of the control parameters according to (4.50) and (4.51) are:

$$\begin{aligned} k_\omega &= 0,1283 \\ k_{pd} &= 11,3585 \end{aligned} \tag{4.53}$$

This system is stable, as k_ω is bigger than the critical one, and the dynamics will be as desired because the double poles dominate the third one.

$$\begin{aligned} k_\omega &> k_\omega^{crit} = 0,0068m/s \\ \frac{\sigma_3}{\sigma_d} &= 6,3 \end{aligned} \tag{4.54}$$

This can be seen in the following root locus, with the poles for the chosen k_{pd} in black. The step response is shown in Figure 4.7.

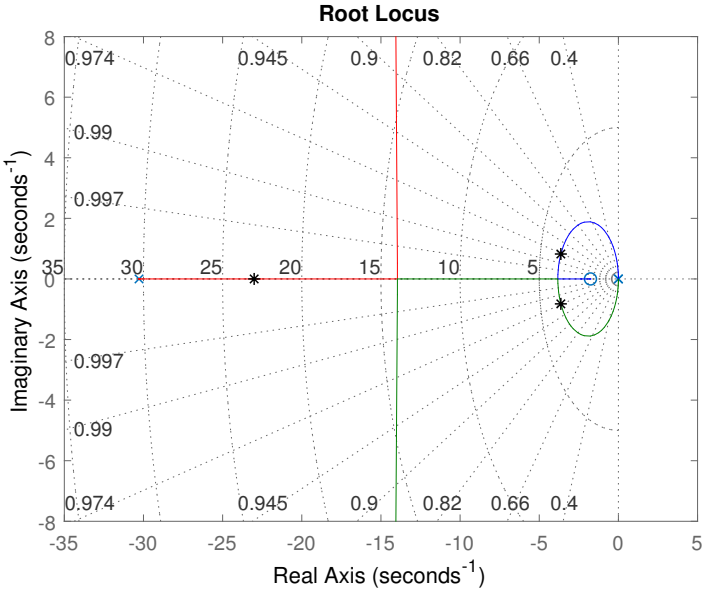

Figure 4.6: Root locus of the distance system with feedforward and the chosen control parameters

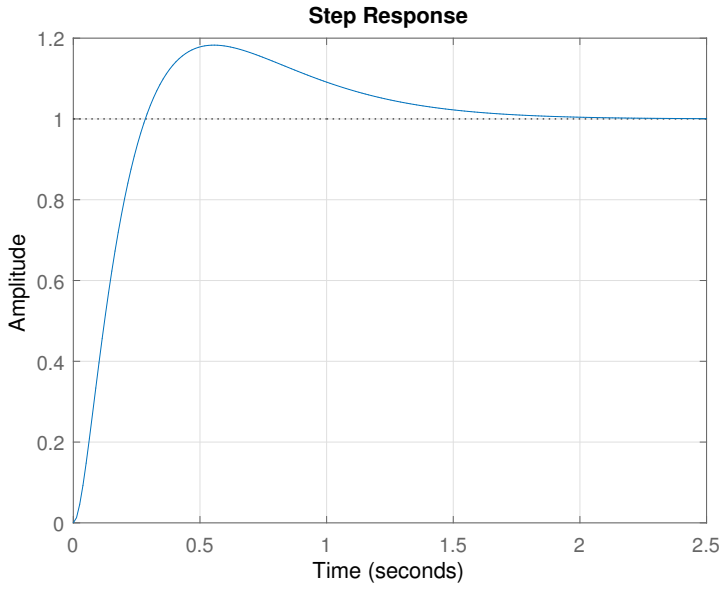

Figure 4.7: Unit step response of the distance system with feedforward and the chosen control parameters

As in the case of the velocity, M_p is different than the one used for the calcu-

lations. That is again because of the zero of the system.

This problem of this system is that it is very sensitive to a change of the desired velocity. For the same desired M_p and t_S, a little change in the velocity can produce either that the necessary k_ω is too small and the system becomes unstable, or that the double poles stop being dominant because the third one is too close to them in the s space. The following root locus serve as an example of that.

Figure 4.8 shows the poles given by making the same calculations previously explained, but with a slightly faster velocity, resulting in a obvious unstable system. Figure 4.9 illustrates the same but for a slower velocity. In this case, the system stays stable, but the double poles have lost their dominance and the response of the system isn't what has been designed.

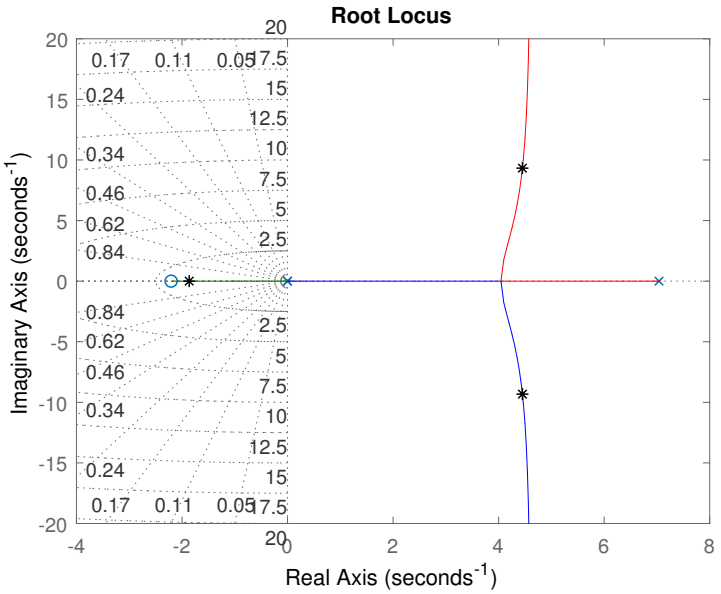

Figure 4.8: Root Locus for the chosen t_s and M_p but $v_d = 0,15m/s$

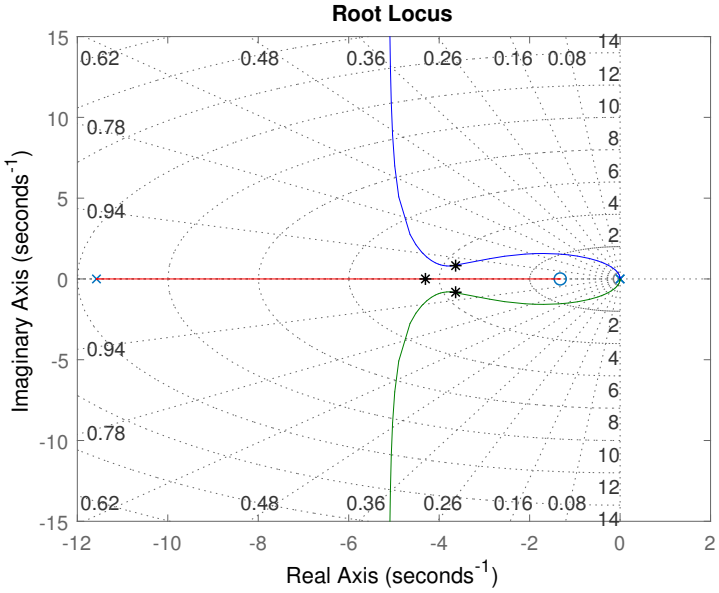

Figure 4.9: Root Locus for the chosen t_s and M_p but $v_d = 0,09m/s$

Changing the setting time of the maximum overshot performance will have a similar effect to the system.

In this case, all the calculations have been done with the B_f calculated in Section 5.2, not the one calculated in Section 4.2.1 to make the system stable. Thanks to the feed forward term, the value of B_f does not affect at all the response of the system, as k_ω will have the necessary value to compensate it. In summary, if B_f were to be changed in the described system, the calculated value of k_ω would be different, but the resulting poles and response would be the same.

As recalculating k_ω and k_{pd} for each case is proved to be a bad control strategy, the control parameters will be fixed with values that ensure the stability and relatively good response of the system within a chosen range of velocities. If this range is $v_d \in (0; 0,3]m/s$, the fixed value of k_ω must be higher than the critical one for the maximum velocity, i.e. 0,3 m/s. The chosen value is 1,5 times the critical velocity.

$$k_\omega = 1.5k_\omega^{crit}(v_d = 0,3) = 0,0271m/s \tag{4.55}$$

For this value of k_ω, k_{pd} is calculated with (4.51) for $t_s = 1s$ and $M_p = 0,01$

$$k_{pd} = 5,7149 \tag{4.56}$$

The poles of the resulting system and its unit step response are illustrated in the following figures.

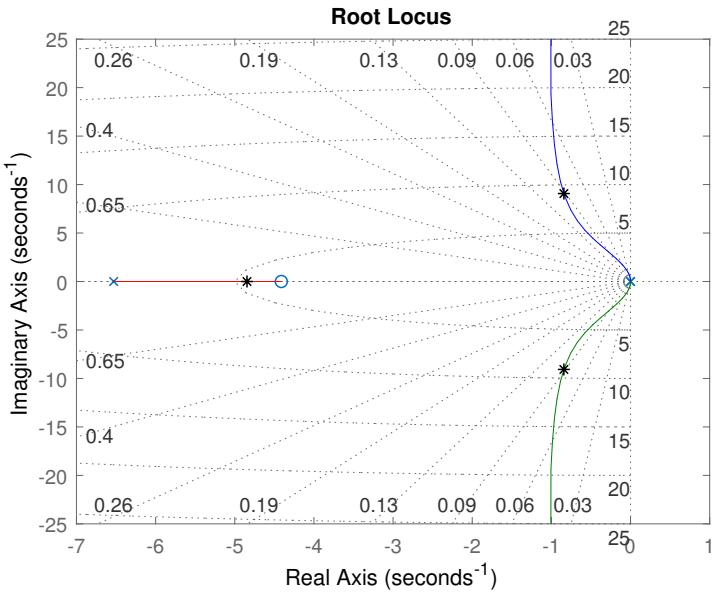

Figure 4.10: Root Locus of the final distance subsystem for $v_d = 0,3m/s$

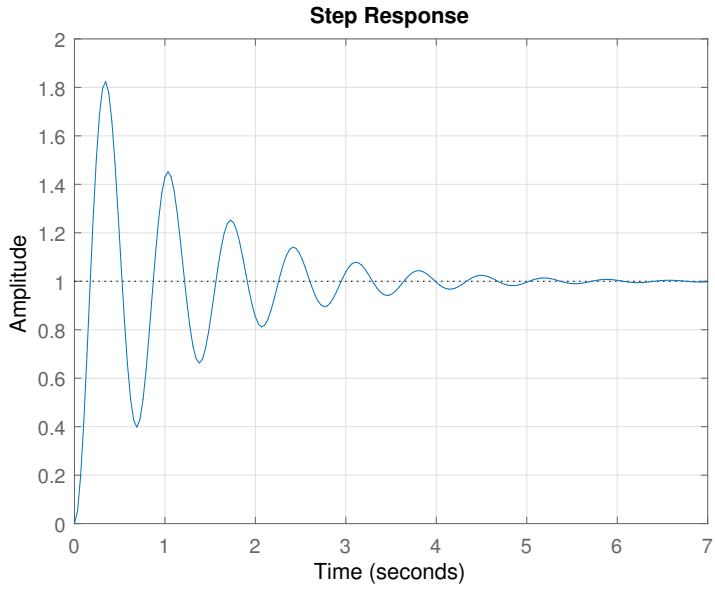

Figure 4.11: Unit step response of the final distance subsystem for $v_d = 0,3m/s$

As k_ω is fixed, the shape of the root locus is also fixed and the dynamics of the system are not as expected from the equations. The response time is approximately 4,5 seconds and the maximum overshot performance is 0,8 per unit, which can also be due the zero of the system.

The following graphics show a comparative of the poles and the response of the system for the chosen fix k_ω and k_{pd} and different velocities within the chosen range.

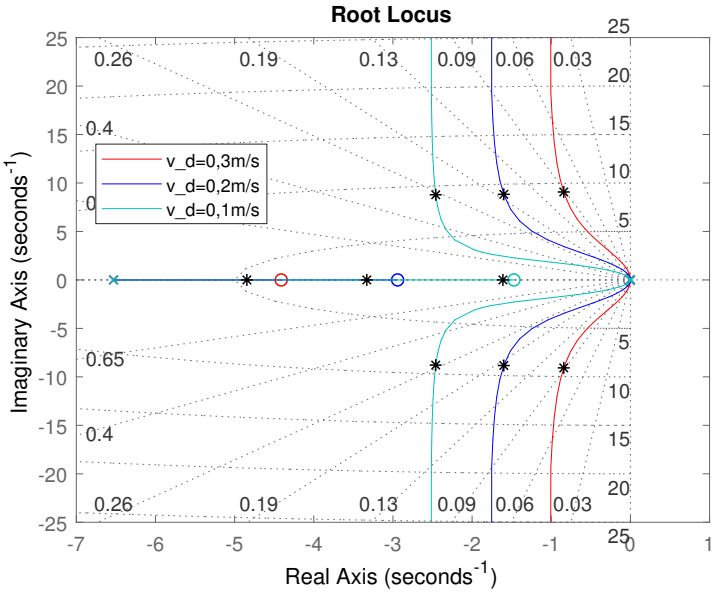

Figure 4.12: Root Locus of the final distance subsystem for different speeds

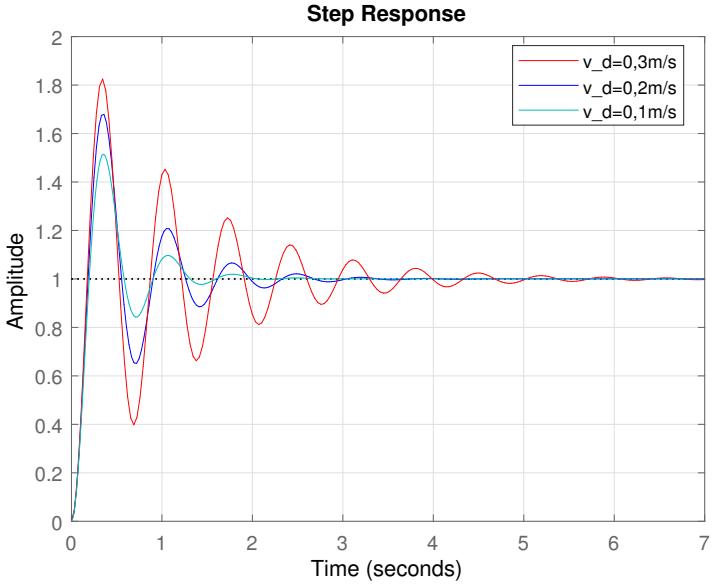

Figure 4.13: Unit step response of the final distance subsystem for different speeds

As seen in the figures above, the response time of the system improves as the the velocity decreases, due to the root locus moving to the left. However, this change of the root locus has a negative effect: the zero will also move closer to the imaginary axis for a smaller velocity, which will make the double poles become less dominant. As the single pole becomes slower, the the behaviour of the system becomes more difficult to predict.

Chapter 5

Estimation of the Parameters of the Robot

In order to run the simulations, all the parameters that appear in the equations have to be measured in the real life prototype. Some of the parameters have been measured directly in the prototype, while others have been estimated from known properties.

5.1 Geometrical Parameters

The situation of the Center of Mass has been estimated experimentally by making the robot stand balanced with its only support on that point. All of the subsequent length parameters have been measured according to it.

To estimate the Moment of Inertia in the z axis, the robot is considered a simple rectangular prism with its mass, length and width. The corresponding formula is:

$$I_z = \frac{1}{12} m \left(W^2 + L^2\right) \tag{5.1}$$

The values of all these parameters are found in Table 5.2.

5.2 Viscous Friction Coefficient

The viscous friction coefficient has been estimated using the specifications of the motor provided by the manufacturer [11].

Free-run speed	ω_{mfr}	1030 rpm	107,861 rad/s
Free-run current	I_{mfr}	300 mA	0,3 A
Stall current	I_{mstall}	5600 mA	5,6 A
Stall torque	τ_{mstall}	44 oz in	0,310708 Nm

Table 5.1: Some of the values of the General Specifications of the Gearmotor (at 12V)

The constant of the motor (K_m) is the relation between the torque and the given current, i.e.:

$$K_m = \frac{\tau_{mstall}}{I_{mstall}}, \tag{5.2}$$

and its numerical value is:

$$K_m = 0,05548 Nm/A \tag{5.3}$$

Having found K_m, B_f can be calculated by applying the Conservation of Angular Momentum for the free-run of the motor.

$$0 = -B_f \omega_{mfr} + K_m I_{mfr} \tag{5.4}$$

$$B_f = \frac{K_m I_{mfr}}{\omega_{mfr}} \tag{5.5}$$

The value of B_s is found in Table 5.2.

5.3 Summary of Parameter Values

The following table contains all of the measured and calculated values mentioned above. Unless it is specified otherwise, these are the values that have been used in the succeeding simulations.

Parameter	**Meaning**	**Measurement [SI]**
a	Half of the distance between the wheels	0,085
b	Distance from the Mass Center to the axle	0,030
l	Distance from the axle to the tracking sensor	0,068
r	Radius of the driving wheels	0,035
L	Length	0,250
W	Width	0,180
m	Mass	1,175
I_z	Moment of inertia in the z axis	$9,292\ 10^{-3}$
B_f	Viscous Friction Coefficient of the Motors	$1,543\ 10^{-4}$

Table 5.2: Experimental Values of the Parameters

Chapter 6

Simulations

Both the nonlinear model found in Chapter 2 and its linearization from Chapter 3 have been simulated using Matlab and Simulink.

The controllers used are the ones defined in Chapter 4 and the values of the parameters are the ones from Table 5.2.

6.1 Simulations Using the Linearized Model

The linear system has been simulated by using the space state description (Equations (4.34)). As in this case the two subsystems are interconnected, the system does not behave exactly as the decoupled system. In this case, if one of the systems became unstable, the other one would lose its stability, too. Nevertheless, for the case with $c = 0$ and a stable system the response should be exactly the same. The designed control algorithm guarantees the stability for $v_d \in (0; 0, 3]m/s$

As an example of this, Figure 6.1 shows the response of the linear system

simulation for $v_d = 0,15m/s$ and the following initial conditions

$$\begin{aligned} d &= 0,05m \\ \theta_e &= 0rad \\ v_x &= 0m/s \\ \omega &= 0rad/s, \end{aligned} \tag{6.1}$$

while Figure 6.2 shows the response for the two subsystems separately. It is clear in the images that the results are exactly the same.

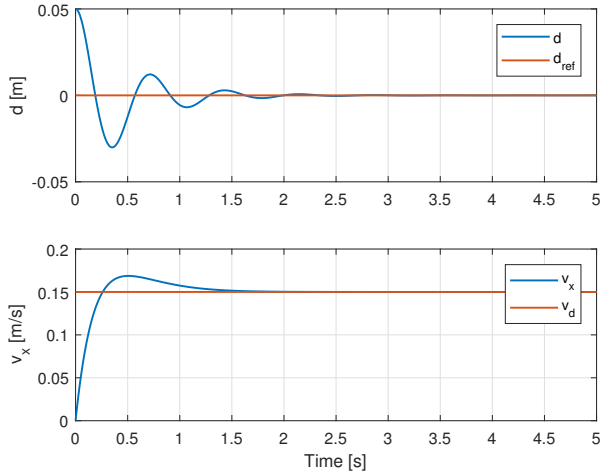

Figure 6.1: Response of the linear system for $v_d = 0,15m/s$

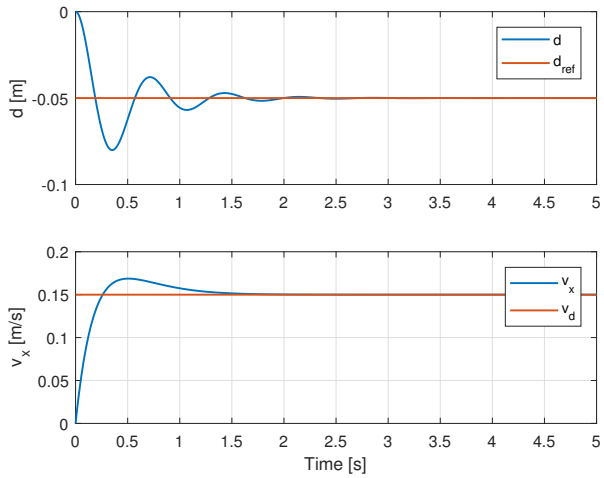

Figure 6.2: Response of the linear system for $v_d = 0,15m/s$ for the decoupled system

Figure 6.3 shows the response of the simulation of the linear system for $v_d = 0,2m/s$ and the following initial conditions, which are different from the ones of simple step responses for the separated systems.

$$\begin{aligned} d &= 0,05m \\ \theta_e &= 0.1rad \\ v_x &= 0.1m/s \\ \omega &= 0rad/s \end{aligned} \tag{6.2}$$

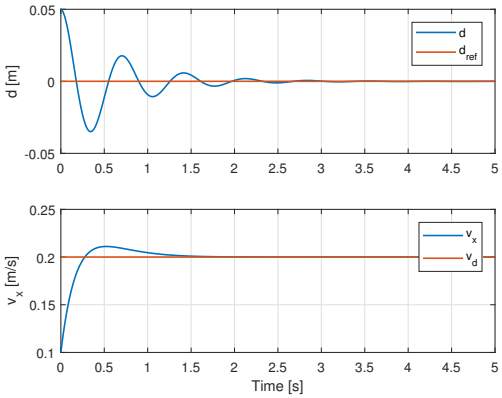

Figure 6.3: Response of the linear system for $v_d = 0,2m/s$

Figure 6.4 shows the original control parameters together with the defined ones, as an illustration of how the subsystems affect each other. τ_R and τ_L show oscillation because they depend on both u_d and u_v and the distance subsystem oscillates.

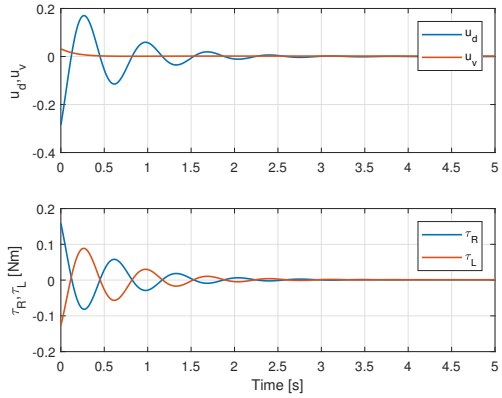

Figure 6.4: Control parameters of the linear system for $v_d = 0,2m/s$

Another interesting thing of that simulation is that it allows to have a look at the values of the other state variables (θ_e and ω) which should become zero because the track to follow is a line, i.e. for the stationary state, the robot should be positioned along the line and not rotate. As can be seen in Figure 6.5, this happens as predicted.

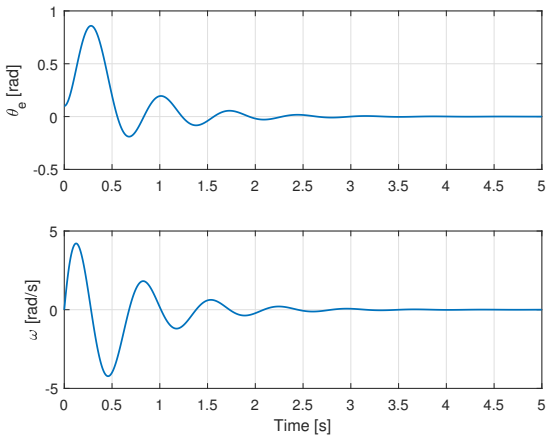

Figure 6.5: Other parameters of the linear system for $v_d = 0,2m/s$

If the curvature is not zero, but constant, i.e., the line to follow is a circle, the response of the system is not so easy to predict. The following figures show the response of the system to the same v_d and initial conditions as above, but $c = 2$.

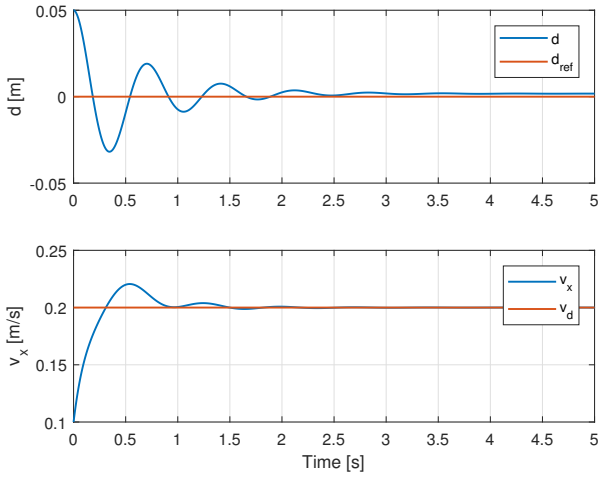

Figure 6.6: Response of the linear system for $v_d = 0,2m/s$ and $c = 2$

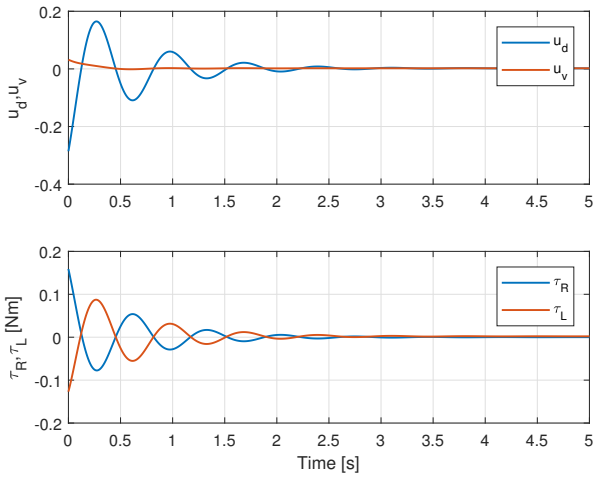

Figure 6.7: Control parameters of the linear system for $v_d = 0,2m/s$ and $c = 2$

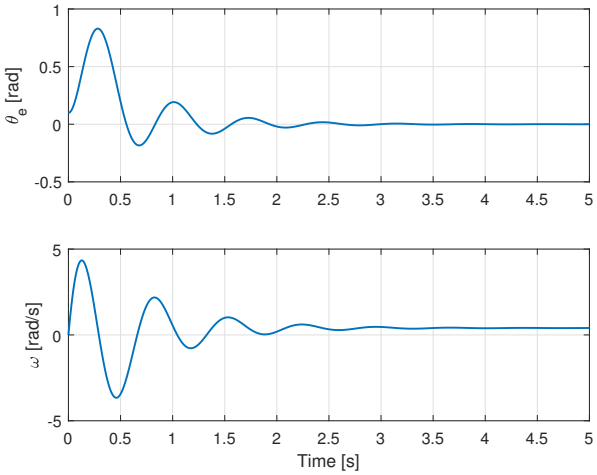

Figure 6.8: Other parameters of the linear system for $v_d = 0,2m/s$ and $c = 2$

The system stays stable and with a similar response, but, for this case, the stationary error is not zero, as seen in Figure 6.6. As expected, the angle has to change to adapt to the curve and the angular velocity on the stationary obtains the necessary value to follow the track (see Figure 6.8).

6.2 Simulations Using the Nonlinear Model

The nonlinear system simulates the original equations (2.39) and it should behave exactly like the linear one around the equilibrium values. Figure 6.9 shows the response for both the linear and the nonlinear model for $v_d = 0,15m/s$ and initial conditions very close to the equilibrium. As seen, the response of the non-linear system is very close to the linear one.

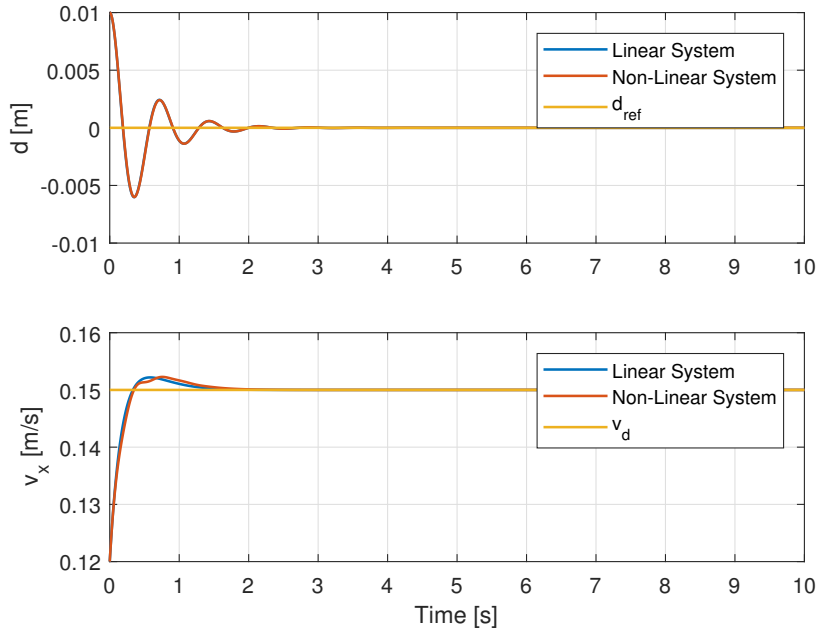

Figure 6.9: Comparative between the linear and the non-linear system (equilibrium)

The following figures show the behaviour of the nonlinear system for the same two cases done in last section and in comparison to it.

The figures show that, even the response is not exactly the same, for $c = 0$, the system arrives to the same stationary value. On the other hand, for $c = 2$, even the response is very similar, θ_e doesn't become zero.

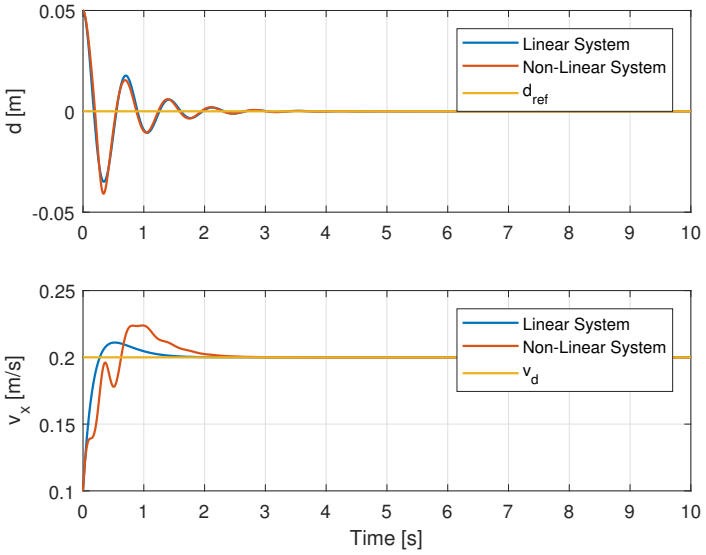

Figure 6.10: Response of the linear and the nonlinear system for $v_d = 0,2$ and $c = 0$

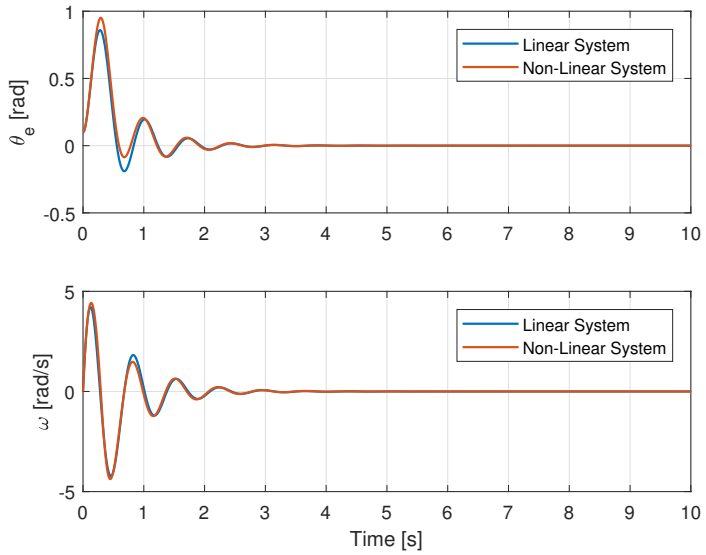

Figure 6.11: Other parameters of the linear and the nonlinear system for $v_d = 0,2$ and $c = 0$

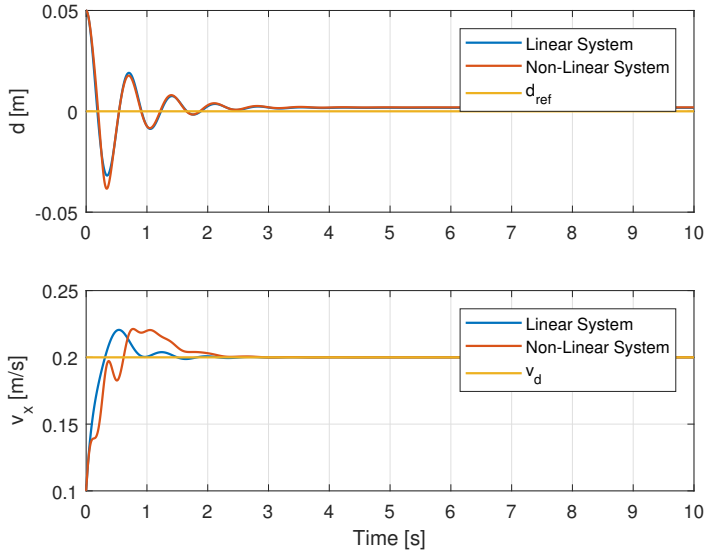

Figure 6.12: Response of the linear and the nonlinear system for $v_d = 0,2$ and $c = 2$

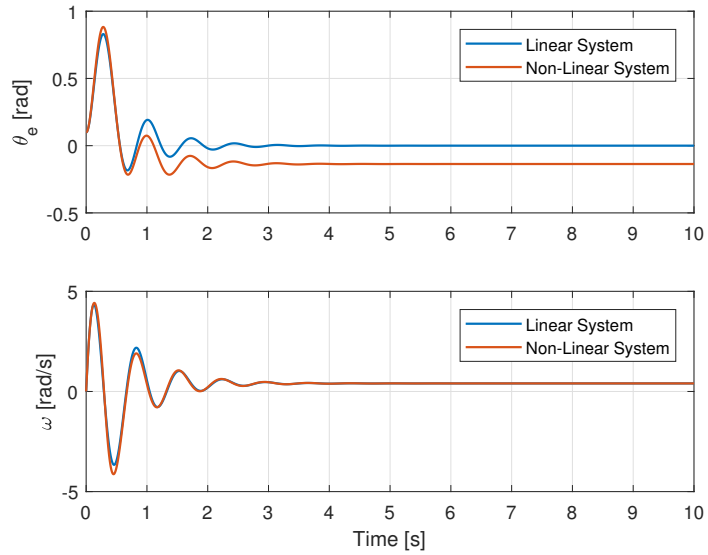

Figure 6.13: Other parameters of the linear and the nonlinear system for $v_d = 0,2$ and $c = 2$

Conclusions

The objectives have been accomplished and a control algorithm using a complete mechanical model has been designed for regulating both the distance to the tracking line and the longitudinal speed of the robot. More specifically:

- A complete mechanical model has been developed and its linear approximation has been obtained in order to design the control.
- The linear system can be decoupled for a particular case (straight line), which gives two subsystems: one that controls the velocity and one that controls the distance to the tracking line.
- The velocity subsystem can be easily controlled with a PI controller independent of the desired velocity of the robot.
- The distance subsystem is unstable for high velocities, so a feed forward control damping the angular speed has been implemented. The resulting controller works within a certain chosen range of velocities.
- The instability of the distance subsystem might be caused by an underestimation of the viscous friction coefficient of the motors. In spite of that, the damping allows to stabilize the system for any value of that coefficient.
- Simulations for both the linear system (decoupled and not) and the non linear one have been successfully run and the designed control has been deemed to be suitable for the system.

Several paths of development can be set from this work. The most important one is to put it in practise, i.e., to try the designed control in the actual robot. In order to do that, the value of the angular speed has to be available, either measuring it directly or using a state-observer.

Another possible development is to design an experiment to measure the value of the friction coefficient of the motor, and even to measure the effect of the other present friction forces that might be affecting the system more than has been estimated, and could be added to the model.

Acknowledgements

I would like to thank Arnau Dòria, the advisor of this Bachelor's Thesis, for all of his assistance and guidance during the development of this project.

Appendix A

Budget

This appendix shows the budget breakdown for the present thesis.

A.1 Labour costs

As this is a Bachelor Thesis developed under an academic environment, the student has not earned any salary. In case that the labour hours were paid, this part of the budget would be:

Concept	**Hourly rate[€/h]**	**Units** (h)	**Cost**(€)
Research, documentation and tool learning	40	80	3200
Simulation and development	40	160	6400
Writing and consolidation of data	40	120	4800
Subtotal		360	14400
VAT(21%)			3024
Total			17424

Table A.1: Labour costs

A.2 Development tools and office material

The main material used to develop this project is a laptop computer and the required software, in this case, MATLAB. The university provides free access to literature material, so this cost is not included.

Concept	**Unit cost[€/ut]**	**Units**	**Cost(€)**
Personal computer	800	1	800
MATLAB®2017 (Academic use)	500	1	500
Subtotal			1300
VAT(21%)			273
Total			1573

Table A.2: Development costs

A.3 Total cost

The following table shows the total budget of the project.

Item	**Cost[€]**
Labour	17424
Development	1573
Total (VAT included)	**18997**

Table A.3: Total cost

Bibliography

[1] I. Prats Martinho, *Control design and implementation for a line tracker vehicle*, Bachelor's Thesis, Jun. 2016.

[2] A. Riera Seguí, *Disseny i implementació d'un sistema de comunicacions wifi per a una xarxa de vehicles autònoms*, Bachelor's Thesis, Jun. 2016.

[3] A. Costa, *Design of controllers and its implementation for a line tracker vehicle*, Bachelor's Thesis, Jun. 2017.

[4] J. O. Torta Valmaña, *Hardware design and implementation of two-wheleed vehicle robots for a platooning system*, Bachelor's Thesis, Jun. 2017.

[5] C. Conejo Barceló, *Adaptative cruise control, an scaled model: Platooning using two-wheeled robots*, Bachelor's Thesis, Jun. 2017.

[6] M. K. Ghezzi, *Control of a four in-wheel motor drive electric vehicle*, Master's Thesis, Sep. 2017.

[7] J. Agulló i Batlle, *Mecànica de la partícula i del solid rígid*, 3rd. Publicacions OK PUNT, 2002.

[8] R. Rajamani, *Vehicle Dynamics and Control*. Springer, 2006, ISBN: 9780387263960.

[9] S. A. Velinsky, "Dynamic model based robust tracking control of a differentially steered wheeled mobile robot", *Proceedings of American Control Conference*, Jun. 1998.

[10] K. Ogata, *Modern Control Engineering*, 5th. Pearson, 2010, ISBN: 9780136156734.

[11] *Metal gearmotor 25dx48lmm hp 12v with 48 cpr encoder*, Accessed: 22 of February 2018, Pololu Robotics and Electronics. [Online]. Available: `https://www.pololu.com/product/3214/specs`.